1. Give regular expressions generating the following languages:

a) {w : w contains exactly two 0s}

1*01*01*

b) {w : w contains at least two 0s}

$\Sigma^*0\Sigma^*0\Sigma^*$, here note that $\Sigma = (0 \cup 1)$

c) {w | w begins with a 1 and ends with a 0}

$1\Sigma^*0$

d) {w | w contains at least three 1s }

$\Sigma^*1\Sigma^*1\Sigma^*1\Sigma^*$

e) {w | the length of w is at most 5}

$(\Sigma \cup \varepsilon)(\Sigma \cup \varepsilon)(\Sigma \cup \varepsilon)(\Sigma \cup \varepsilon)(\Sigma \cup \varepsilon)$

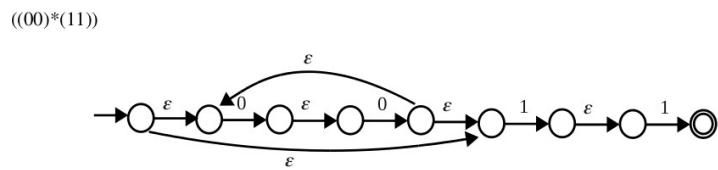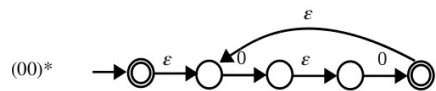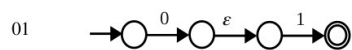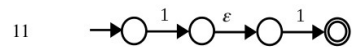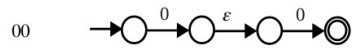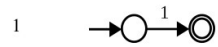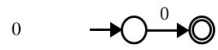**f)** {w | w contains an even number of 0s, or contains exactly two 1s}

Even number of 0s: (1*01*01*)*

Contains Exactly two 1s: 0*10*10*
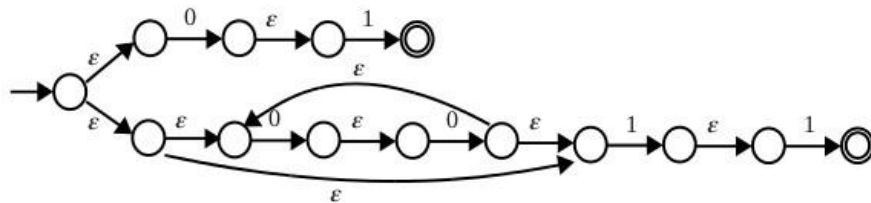
Even number of 0s, or contains exactly two 1s: (1*01*01*)* U 0*10*10*

2.

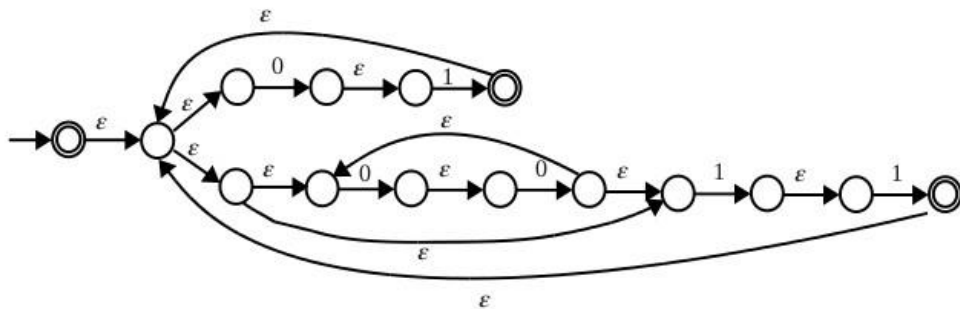**a)** Convert the regular expression (((00)*(11)) ∪ 01)* to an NFA.
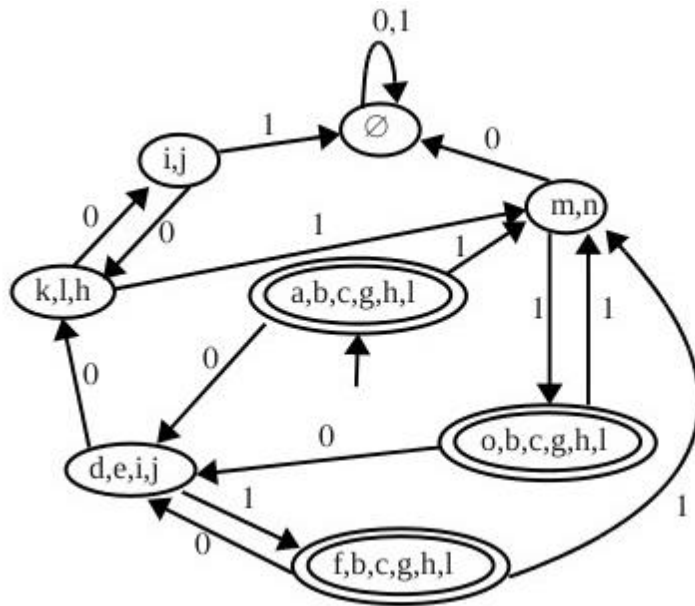
0

1

00

11

01

(00)*

((00)*(11))

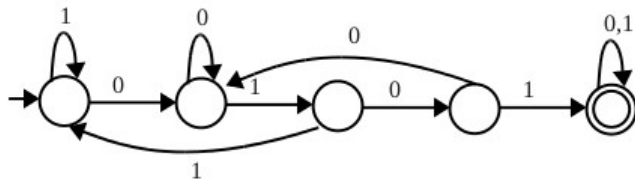((00)*(11)) ∪ 01

(((00)*(11)) ∪ 01)*

b) Convert the NFA (in part a)) to an equivalent DFA. Give only the portion of DFA that is reachable from the start state.
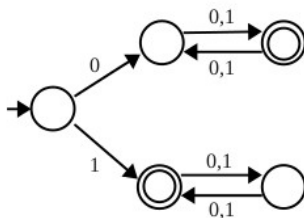

3.

Give state diagrams of DFAs recognizing the following languages. In all parts, the alphabet is {0,1}.
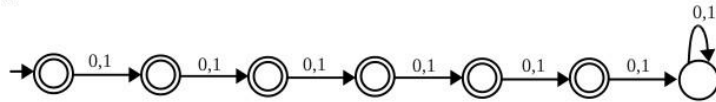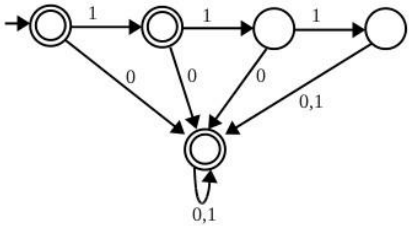
a) {w | w contains the substring 0101.

b) {w | w starts with 0 and has odd length, or starts with 1 and has even length} .

c) {w | the length of w is at most 5}.
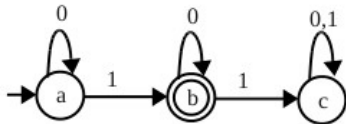
d) {*w* | *w* is any string except 11 and 111}.

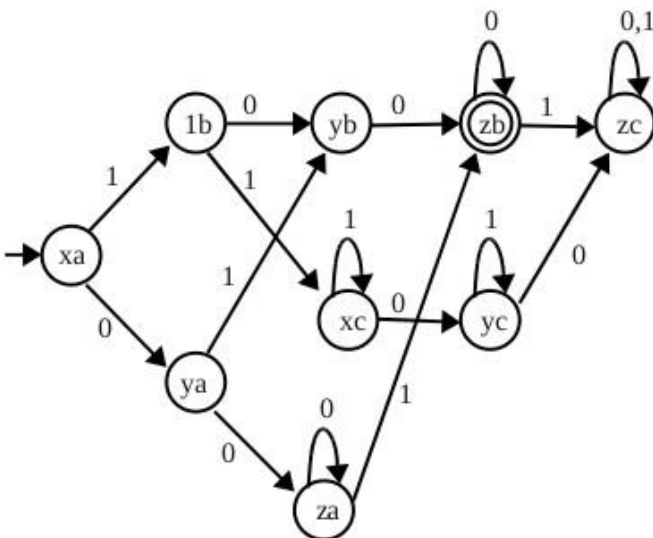**e)** {*w* | *w* contains at least two 0s and at most one 1}.

First draw the DFAs for strings consisting of at least two 0s.:
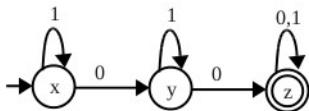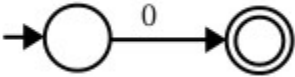

Strings consist of exactly one 1:

Now draw the final DFA which simulates these two DFAs and accepts only when both of these DFAs are in an accept state:
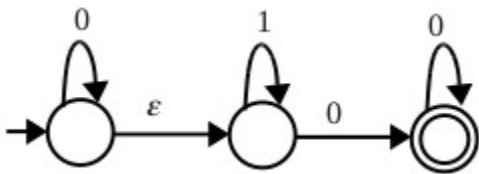
4. Give state diagrams of NFAs with the specified number of states recognizing each of the following languages. In all parts, the alphabet is {0,1}.

a) The language {0} with two states.

b) The language 0*1*0⁺ with three states.

c) The language 1*(001⁺)* with three states